Int. J. on Recent Trends in Engineering and Technology, Vol. 9, No. 1, July 2013

Automatic Feature Subset Selection using Genetic Algorithm for Clustering

A. Srikrishna[1], B. Eswara Reddy[2], V. Sesha Srinivas[1]
[1]Deparment of Information Technology, Rayapati Venkata Ragarao and Jagarlamudi Chndramouli College of Engineeing, Guntur, A.P., India.
vangipuramseshu@gmail.com, atlurisrikrishna@yahoo.com
[2]Department of Computer Science and Engineering, Javaharlal Nehru Technological University, Ananthapur, A.P., India.
eswarcsejntu@gmail.com

Abstract: **Feature subset selection is a process of selecting a subset of minimal, relevant features and is a pre processing technique for a wide variety of applications. High dimensional data clustering is a challenging task in data mining. Reduced set of features helps to make the patterns easier to understand. Reduced set of features are more significant if they are application specific. Almost all existing feature subset selection algorithms are not automatic and are not application specific. This paper made an attempt to find the feature subset for optimal clusters while clustering. The proposed Automatic Feature Subset Selection using Genetic Algorithm (AFSGA) identifies the required features automatically and reduces the computational cost in determining good clusters. The performance of AFSGA is tested using public and synthetic datasets with varying dimensionality. Experimental results have shown the improved efficacy of the algorithm with optimal clusters and computational cost.**

Key words: **feature subset selection, Genetic Algorithm and clustering.**

I. Introduction

Clustering is an unsupervised process of grouping objects into classes of similar objects. A cluster is a collection of objects with high similarity and is dissimilar, to the objects belonging to other clusters [1-2]. Clustering is useful in many applications such as pattern-analysis, grouping, decision-making, and machine-learning situations, including data mining, document retrieval and image segmentation [3-4]. Hierarchical and Partitional are the two well known methods in clustering. Hierarchical methods construct the clusters by recursively partitioning the objects while the partitioning methods divide a dataset with or without overlap [5-6].

One of the challenges of the current clustering algorithms is dealing with high dimensional data. The goal of the feature subset selection is to find a minimum set of features such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all features [7]. Mining on a reduced set of features has an additional benefit. It reduces the number of features appearing the discovered patterns, helping to make the patterns easier to understand [7].

Most feature selection algorithms are focused on heuristic search approaches such as sequential search [8], non linear optimization [9], and genetic algorithms. Basic heuristic methods of attribute subset selection include Stepwise forward selection, backward elimination, combination, and decision tree induction. Stepwise forward selection starts with an empty set of attributes as the reduced set. The best of the original attributes is determined and added to the reduced set. At each subsequent iteration or step, the best of remaining original attributes is added to the set. Stepwise backward elimination starts with the full set of attributes at each step; it removes the worst attribute remaining in the set. Combination of forward selection and back ward elimination selects the best attributes and remove the worst form among the remaining attributes. Decision tree induction algorithms such as ID3, C4.5, and CART, were originally intended for classification. Decision tree induction constructs a flow chart like structure where each internal (non leaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the "best" attribute to partition the data into individual classes [7]. Ferri et. al. have proved that Sequential Floating Forward Search (SFFS) algorithm was the best among the sequential search algorithms [10]. These methods provided solution for feature selection as a supervised learning context, and solutions are evaluated using predictive accuracy[11]. Among these different categories of feature selection algorithms the genetic algorithm is a recent development [12].

Genetic algorithm approach for feature subset selection appears first in 1998[13]. The GA is biologically inspired evolutionary algorithm. It has a great deal of potential in scientific and engineering optimization or search problems [14]. GA can be applicable to feature selection since the selection of subset of features is a search problem. The performance of GA and classical algorithms have compared by Siedlecki and Sklansky [15]. Many literatures were published showing the advantages of GA for Feature Selection [16, 17]. An unsupervised learning via evolutionary search for feature selection is proposed in 2000 [18]. The authors have used an evolutionary local selection algorithm to maintain a diverse population of solutions in multidimensional objective space. Il-Seok Oh et.al. have concluded that no serious attempts have been made to improve the capability of GA and they have developed Hybrid Genetic Algorithms for Feature Selection by embedding the problem specific local search operations in a GA. In their work, a ripple factor is used to control the strength of local improvement and have shown the supremacy

DOI: 01.IJRTET.9.1.560

of GA compared to other algorithms in Feature Subset Selection [12]. Feng Tan et. al. have proposed a mechanism to apply existing feature selection methods on a dataset. The feature subsets from these methods are selected as population in GA [19]. A new genetic algorithm based wrapper feature selection method for classification of hyper spectral image data using SVM is proposed [20]. But for all these algorithms number of features is a priori i.e. the algorithms are not automatic.

More relevant subset of features can be selected if the selection process aimed to application. This paper proposes an Automatic Feature Subset Selection using Genetic Algorithm (AFSGA) for Clustering, which determines subset of features automatically while clustering. A new chromosome representation is modelled for the problem of feature subset selection. The proposed algorithm contains two phases, selection of optimal initial seeds are determined in the first phase and later deals the process of feature subset selection while clustering by selecting CS measure as the fitness function. Efficiency of the algorithm is studied by selecting various public and synthetic datasets. The following sections are divided into scientific background, Genetic Algorithm, Automatic Feature Subset Selection using Genetic Algorithm (AFSGA), experimental results and conclusion.

II. Scientific Background

A. Problem Definition

A data object can be distinguished from others by a collective set of attributes called features, which together represent a pattern [12]. Let $P = \{P_1, P_2,...,P_n\}$ be a set of n data points, each having d features. These patterns can also be represented by a data matrix X_{nxd} with n d-dimensional row vectors. The i^{th} row vector X_i characterizes the i^{th} object from the set P, and each element $X_{i,j}$ in X_i corresponds to the j^{th} feature (j = 1,2,...,d) of the i^{th} data object (i=1,2,...,n).

$$\begin{bmatrix} X_{11} & \cdots & X_{1f} & \cdots & X_{1d} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ X_{i1} & \cdots & X_{if} & \cdots & X_{id} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ X_{n1} & \cdots & X_{nf} & \cdots & X_{nd} \end{bmatrix}$$

Given such an $X_{n\times d}$ matrix, a partitional clustering algorithm tries to find a set of partitions $C = \{C_1, C_2,...,C_K\}$ of K classes, such that the similarity of the data objects in the same cluster is maximum and data objects from different clusters differ as far as possible. The partitions should maintain three properties [21].

1) Each cluster should have at least one data object assigned,

$$\text{i.e., } C_i \neq \Phi \ \forall i \in \{1,2,...,K\}.$$

2) Two different clusters should have no data object in common,

$$\text{i.e., } C_i \cap C_j = \Phi \ \forall \ i \neq j \text{ and } i,j \in \{1,2,...,K\}.$$

3) Each data object should be attached to exactly one cluster only i.e.

$$\bigcup_{i=1}^{k} C_i = P$$

B. Similarity Measure

The dissimilarity between the objects can be computed based on the distance between each pair of objects. The most popular distance measure is Euclidean distance. The Euclidean Distance between objects X_i and X_j is given by:

$$d(X_i,X_j) = \sqrt{\sum_{k=1}^{n} (X_{ik} - X_{jk})^2}$$

where X_{ik} and X_{jk} are the k^{th} coordinates of X_i and X_j respectively.

III. Genetic Algorithm

Genetic Algorithm is one of the nature inspired stochastic evolutionary optimization algorithm can produce competitive solutions for a wide variety of problems [22]. GA maintains a set of solutions called population. Each solution vector is a chromosome. Biological evolution is a process of selecting survival individuals for the next generation. Survival individuals are the fittest chromosomes those are generated from the crossover and mutation genetic operations. Having found the candidate solutions (parents) the crossover takes place, where the parent's genetic information involved in generating new offspring (children individual) [22]. The mutation operation is applied to the offspring population, according to a very small probability; some of the new individuals will suffer mutation (a random and small change to its genetic material information). A new fitness value is calculated to the individuals that have suffered mutation. The generations will be continued with the calculation of new offspring till a stopping criterion is checked [23]. The criterion is a certain value of the fittest chromosome in the population or a maximum number of generations or processing time elapsed.

IV. Automatic Feature Subset Selection Using Genetic Algorithm For Clustering

Automatic Feature Subset Selection using Genetic Algorithm (AFSGA) proposes a Genetic Algorithm based feature selection method to cluster the data. The method contains two steps; first step finds the optimal initial centroids using CS measure. Finding optimal clusters while selecting features based on GA is the second step. First step selects n/10 sets of centroids randomly and selects the best using CS measure, where n is number of elements in dataset. Second step constructs a GA based algorithm to find the minimal set of required features for clustering.

DOI: 01.IJRTET.9.1.560

A. *Selection of Optimal Initial Centroids*

AFSGA determines optimal initial centroids by running k-means algorithm n/10 times.

```
Algorithm InitCentroids(DATA dataset, K no. Of clusters)
{
        Min=0;
        for i=1 to n/10
        {
                C = k-means(DATA,K);
                csm=CSMeasure(C,DATA);
                if(min>csm)
                {
                        min=csm;
                        initcentroid=C;
                }
        }
        return initcentroid
}
```

The procedural steps and representation of the chromosome to determine the necessary features from the given dataset are as follows.

B. *Chromosome Representation*

Single chromosome is represented by a single bit vector structure. A vector of size D (number of features) is the chromosome in AFSGA. Each bit represents the activation status of the feature. One indicates the feature activation while zero states the inactiveness in the clustering process. Each bit is a gene, a set of genes makes a chromosome which represents a set of features necessary for clustering. The conceptual model of the chromosome is in the fig No1.

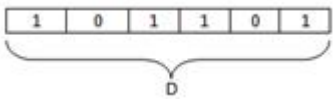

Fig. 1. Chromosome Representation

In the figure the chromosome is with D (6) features, among all the 1,3,4 and 6 features are active for clustering.

A GA based algorithm contains a set of solutions (chromosomes) called population. Here the population size is selected as n number of data objects.

B. *Population Initialization*

Each bit is initialised either with one or zero based on a generated random number. If the random number is lesser than 0.5, set the bit value as one otherwise set to zero. The algorithm is as follows:

```
Algorithm InitPop( D dataset size)
{
    for ( i = 1 to n )
        for ( each feature f in i^th chromosome )
        {
            if (rand (1) < 0.5 )
                f=1;
            else
                f=0;
        }
}
```

The sample population generated by the above algorithm with D (6) features and n (5) data objects is shown in the following fig. No 2.

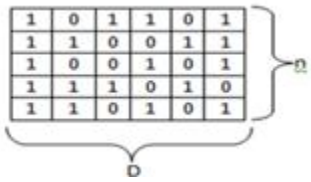

Fig. 2. Sample Population

C. *Selection of Parents*

Parents or candidate solutions are selected randomly from the current population.

```
Algorithm SelectParent(i:CurrentChromosome, p:Population
size)
{
    v=randperm(popsize);
    j=1;
    while j<=2
    {
    if v(1)~=i
        {
        parent(j)=v(1);
        j=j+1;
    }
    v=v(1,2:length(v));
  }
}
```

D. *Crossover and mutation operators*

The principle of applying genetic operators is to change the chromosomes in successive generations until stopping criterion is met. Crossover and mutation operators are the two genetic operations in genetic algorithms. Here the proposed algorithm uses single point crossover as the crossover operation. In single point crossover, two parents are selected randomly from the current population. Select a value v between 1 and D. Form a new offspring by combining the feature bits 1 to v from parent 1 and the feature bits v+1 to d from parent 2. Example for single-point crossover is shown in Fig. No.3

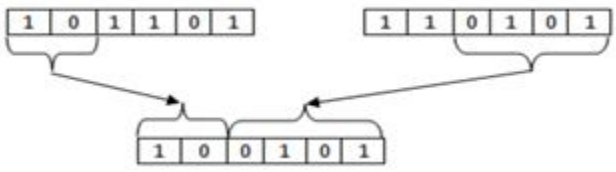

Fig. 3. Single Point Crossover

Mutation operation is applied on each new offspring using the following algorithm. Mutation rate P_m is the input parameter with the value 0.1. The sample mutation is shown in fig. No. 4

DOI: 01.IJRTET.9.1.560

Algorithm Mutation (O offspring, P_m mutation rate)
{
Let n_0 and n_1 be the number of zero bits and one bits respectively in the offspring.
$P_1=P_m$; $P_0=P_m x n_1/n_0$;
for (each feature f in the chromosome)
{
Generate a random number r within the range [0,1].
if (f=1 and r<P_1)
convert f to 0;
else if (f=0 and r<P_0)
convert f to 1;
}
}

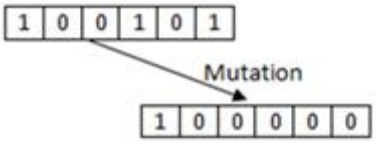

Fig. 4. Mutation Example

E. Fitness Function

The quality of the clustering solution is assessed using cluster validity measures. Most of the cluster validity measures are the ratio of intra cluster distance and inter cluster distance. The proposed algorithm selects CS measure as the fitness function. CS measure is one of the cluster validity measure developed based on compactness and separation of clusters in a clustering solution [24].

Chou *et al*. (2004) have proposed the CS measure for evaluating the validity of a clustering scheme [24]. The centroid of a cluster is computed by averaging the elements that belong to the same cluster using

$$\vec{m}_i = \frac{1}{N_i} \sum_{X_j \in C_i} \overline{X}_j$$

$$CS = \frac{\sum_{i=1}^{k} \left[\frac{1}{N_i} \sum_{\vec{X}_i \in C_i} \max_{\vec{X}_q \in C_i} \{ d (\overline{X}_i , \vec{X}_q)\} \right]}{\sum_{i=1}^{k} \left[\min_{j \in k , j \neq i} \{ d (\vec{m}_i , \vec{m}_q)\} \right]}$$

CS measure is a function of the ratio of the sum of within-cluster distance to between-cluster distance. The cluster configuration that minimizes CS is taken as the optimal solution.

F. Stopping Criterion

The proposed algorithm selects stagnation by convergence as the stopping criterion. The difference of fitness value of fittest individuals in any two successive generations is less than 0.0001 is the convergence criterion for the proposed algorithm.

The AFSGA for feature selection: The AFSGA selects minimum number and relevant features for clustering.
Input: Data set with n objects each contains D number of features, k number of clusters, mutation rate P_m
Output: Best chromosome with limited or minimal number of features.
Procedure:
Step 1: Determining initial centroids
1.1 Select n/10 sets of centroids $C_1,C_2,\ldots C_n$ randomly from the dataset.
1.2 For each data object, find the centroids C_i nearest it. Put the data object in the cluster identified with this nearest centroid.
1.3 Evaluate the quality of each clustering solution obtained in the previous step using CS measure.
1.4 The centroids C_i, with minimum CS measure is selected as the initial centroids
Step 2: Generate initial population of size n using algorithm InitPop.
Step 3: Evaluate fitness of each chromosome using CS measure, fitness function
Step 4: Generate new offspring for each chromosome applying crossover on the selected candidate individuals
Step 5: Apply mutation operation on each new offspring obtained in the previous step
Step 6: Evaluate fitness of each new offspring.
Step 7: Repeat the steps4 to step 6 until difference of fitness value of fittest individuals in any two successive generations is less than 0.0001.

V. Experimental Results

The experiments are conducted on three public datasets with variable sample space.

The real data sets used [25]:
1. Iris plants database (n = 150, D = 4, k = 3): This is a well-known database with 4 inputs, 3 classes, and 150 data vectors. The data set consists of three different species of iris flower: Iris setosa, Iris virginica, and Iris versicolour.
2. Glass (n = 214, D = 9, k = 6): The data were sampled from six different types of glass.
3. Wine (n = 178, D = 13, k = 3): The wine data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three clusters of wines.
4. Synthetic dataset1 (n=450, D=15, k= 3).
5. Synthetic dataset2 (n=850, D=20, k= 3).

The algorithm is executed for 40 times on each dataset. The fittest chromosome is identified in each independent run. The features observed in most of the fittest chromosomes are the minimal features. The average value of the fittest chromosome in 40 independent run and the number of features are tabulated in table 1, comparing with the k-means clustering results. The results demonstrated that the proposed algorithm generates more optimal clustering solution with minimal features.

DOI: 01.IJRTET.9.1.560

TABLE I: PERFORMANCE EVALUATION OF AFSGA

Dataset	Algorithm	CS measure	Number of features	% of data reduced
Iris	K-means	0.1281	4	0
	AFSGA	0.0608	3	25
Wine	K-means	0.2550	13	0
	AFSGA	0.1773	11	15.38
Glass	K-means	0.3733	9	0
	AFSGA	0.4355	8	11.11
Synthetic1	K-means	0.06189	15	0
	AFSGA	0.0361	13	13.33
Synthetic2	K-means	0.0763	20	0
	AFSGA	0.0676	13	35

VI. CONCLUSION AND FUTURE WORK

Feature subset selection is the problem of selecting a subset of features based on some optimization criterion. AFSGA selects subset of features based on CS measure using genetic algorithm for the process of clustering. The results of AFSGA are compared with the classical clustering algorithm. The results have demonstrated the improved efficiency of AFSGA compared to k-means. Differential evolution (DE) is one of the most powerful stochastic real-parameter optimization algorithms in current use, which takes negligible input number of parameters compared to GA [26]. Generating feature subset algorithm using differential evolution is our upcoming work.

REFERENCES

[1] Hansen P, Jaumard P (1997), "Cluster analysis and mathematical programming. Mathematical Programming," vol.79, pp.191-215.

[2] Hartigan J (1975), "Clustering Algorithms" John Wiley and Sons.

[3] Hartigan JA and Wong MA (1979), "A K-means clustering algorithm". *Applied Statistics* vol.28, pp.100–108.

[4] Jain AK and Dubes RC (1988), "Algorithms for Clustering Data, Prentice Hall," ISBN: 013022278X, pp: 320.

[5] Jain AK, Murty MN, Flynn PJ (1999), "Data clustering: a Review. ACM Computing Surveys," vol.31, no.3, pp.264-323.

[6] Jain AK (2010) "Data Clustering: 50 Years Beyond K-Means" , Pattern Recognition letters, 31, pp 651- 666

[7] Han J and Kamber M (2004) "Data mining concepts and techniques", second edition, Morgan Kaufman Publishers.

[8] J. Kittler. Feature selection and extraction. In Y. Fu, editor, Handbook of Pattern Recognition and Image Processing, New York, 1978. Academic Press.

[9] P. S. Bradley, O. L. Mangasarian, and W. N. Street. Feature selection via mathematical programming. INFORMS Journal on Computing, 10(2):209-217,1998.

[10] F.J. Ferri, P. Pudil, M. Hatef, and J. Kittler, "Comparative study of Techniques for Large-Scale feature Selection," Pattern Recognition in Practice IV, E.S. Gelsema and L.N. Kanal, eds., pp. 403-413, 1994.

[11] M. Dash and H. Liu. Feature selection for classi_cation. Intelligent Data Analysis, 1(3):131{156, 1997.

[12] II-Seok Oh, Jin Seon Lee, and Byung-Ro Moon(2004), " Hybrid Genetic Algorithms for Feature Selection", IEEE trans. On Pattern analysis and machine intelligence, vol. 26, No.11, pp.1424-1437.

[13] Jihoon Yang and Vasant Honavar. Feature suset selection using Genetic Algorithm. IEEE Intelligent Systems, 1998.

[14] Swagatam Das, Ajith Abraham and Amit Konar "Metaheuristic Clustering" 2009 Springer-Verlag Berlin Heidelberg, ISBN 978-3-540-92172-1, ISSN 1860949X

[15] W. Siedlecki and J. Sklansky, " A note on Genetic Algorithms for Large-Scale Feature Selection," Pattern Recognition Letters, vol. 10, pp. 335-347, 1989.

[16] M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, and A.K. Jain, " Dimensionality Reduction Using Genetic Algorithms," IEEE Trans. Evolutionary Computation, vol.4, no.2, pp.164-171, July 2000.

[17] J.H. Yang and V. Honavar, "Feature Subset Selection Using a Genetic Algorithm," IEEE Intelligent Systems, vol. 13, no. 2, pp.44-49, 1998.

[18] Y. Kim, W. Street, and F. Menczer. Feature selection for unsupervised learning via evolutionary search. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 365–369, 2000.

[19] Feng Tan, Xuezheng Fu, Yanqing Zhang,and Anu G. Bourgeois. A genetic algorithm-based method for feature subset selection. Soft Computing - A Fusion of Foundations, Methodologies and Applications, Volume 12 , Issue 2 , Pages: 111 – 120, Springer-Verlag, 2007.

[20] Li Zhuo , Jing Zheng, Fang Wang, Xia Li, Bin Ai and Junping Qian. A Genetic Algorithm based Wrapper Feature selection method for Classification of Hyperspectral Images using Support Vector Machine. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B7. Beijing, 2008.

[21] Swagatam Das, Ajith Abraham (2008) "Automatic Clustering Using An Improved Differential Evolution Algorithm", Ieee Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, Vol. 38, No. 1, Pp218-237.

[22] Man, K.F, Tang, K.S. Kwong, S, " Genetic algorithms: concepts and applications [in engineering design", IEEE transactions on Industrial Electronics, 1996, vol. 43, No. 5, pp. 519-534.

[23] W. Siedlecki and j. Skelansky, "A Note on Genetic algorithms for large scale feature selection", Pattern recognition letters, vol. 10, pp. 335-347, 1989.

[24] Chou CH, Su MC, Lai E (2004), "A new cluster validity measure and its application to image compression," Pattern Anal. Appl., vol. 7, no. 2, pp. 205–220.

[25] P.M Murphy and D.W. Aha, " UCI Repository for Machine Learning databases", Technical report, Dept. Of Information and Computer Science, Univ. of California, Irvine, Calif., 1994, http://www.ics.uci.edu/mlearn/MLrepository.html

[26] Swagatam Das, P. Nagaratnam Suganthan,*"Differential Evolution: A Survey of the State-of-the-Art"*, IEEE Transactions On Evolutionary Computation, vol. 15, no. 1, February 2011 pg.4-32.

DOI: 01.IJRTET.9.1.560